DOI:
10.1023/A:1021352309671

Document Version
Peer reviewed version

Citation for published version (APA):
Degtyarev, A., Fisher, M., & Lisitsa, A. (2002). Equality and Monodic First-Order temporal Logic. *Studia Logica*, *72*(2), 147-156. https://doi.org/10.1023/A:1021352309671

Equality and Monodic First-Order Temporal Logic

Anatoli Degtyarev, Michael Fisher, and Alexei Lisitsa

Logic and Computation Group, Department of Computer Science
University of Liverpool, Liverpool L69 7ZF, U.K.
{A.Degtyarev,M.Fisher,A.Lisitsa}@csc.liv.ac.uk

Abstract. It has been shown recently that monodic first-order temporal logic without functional symbols but with equality is incomplete, i.e. the set of the valid formulae of this logic is not recursively enumerable. In this paper we show that an even simpler fragment consisting of monodic monadic two-variable formulae is not recursively enumerable.

1 Introduction

It has been known for a long time that first-order temporal logic over the natural numbers is incomplete [Sza86, SH88]. Thus, there is no finitary inference system which is sound and complete for the logic[1], or equivalently, the set of valid formulae of the logic is not recursively enumerable. Recently, the interesting *monodic* fragment of first-order temporal logic has been investigated [HWZ00], which has a quite transparent (and intuitive) syntactic definition and a finite inference system [WZ01]. Moreover many important subfragments of the monodic fragment turn out to be decidable [HWZ00, WZ01]. Unfortunately, all the positive properties of the monodic fragment concerning completeness and decidability hold only for the logic without equality. For example, in [WZ01] it was shown that the set of valid monodic formulae becomes not recursively enumerable after adding equality. However, the given proof left open some questions concerning the minimum requirements of the monodic language with equality necessary for obtaining incompleteness. Related questions concerning what will happen with decidable fragments of monodic first-order temporal logic, such as monadic or two-variable varieties, once equipped with equality, have also been left open in [HWZ00] and [WZ01].

Below we prove that even the intersection of monodic monadic and two-variable fragments becomes not recursively enumerable once equality is added.

The language $\mathcal{TL}$ of the first-order temporal logic over the natural numbers is constructed in the standard way (see, for example, [Fis97, HWZ00]) from a classical (non-temporal) first-order language $\mathcal{L}$ (with equality but without functional symbols) and a set of future-time temporal operators '$\Diamond$' (*sometime*), '$\Box$' (*always*), '$\bigcirc$' (*in the next moment*), '$\mathcal{U}$'(*until*).

Formulae in $\mathcal{TL}$ are interpreted in *first-order temporal structures* of the form $\mathfrak{M} = \langle D, \mathcal{I} \rangle$, where D is a non-empty set, the *domain* of $\mathfrak{M}$, and $\mathcal{I}$ is a function associating, with every moment of time $n \in \mathbb{N}$, an interpretation of predicate and constant symbols of $\mathcal{L}$ over D. First-order (non-temporal) structures corresponding to each point of time

[1] However there are complete infinitary systems with ω-type rules [Kaw87, Sza87, Sza95].

n will be denoted by $\mathfrak{M}_n = \langle D, \mathcal{I}(n)\rangle$. Intuitively, the interpretations of $\mathcal{TL}$-formulae are sequences of first-order structures, or *states* of $\mathfrak{M}$, such as $\mathfrak{M}_0, \mathfrak{M}_1, \ldots, \mathfrak{M}_n \ldots$. An *assignment* in D is a function $\mathfrak{a}$ from the set $\mathcal{L}_{\mathbf{v}}$ of individual variables of $\mathcal{L}$ to D. If P is a predicate symbol then $P^{\mathcal{I}(n)}$ (or simply P^n if $\mathcal{I}$ is understood) is the interpretation of P in the state $\mathfrak{M}_n$. We require that (individual) variables and constants of $\mathcal{TL}$ are *rigid*, that is neither assignments nor interpretations of constants depend on the state in which they are evaluated.

The *truth-relation* $\mathfrak{M}_n \models^{\mathfrak{a}} \varphi$ (or simply $n \models^{\mathfrak{a}} \varphi$, if $\mathfrak{M}$ is understood) in the structure $\mathfrak{M}$ for the assignment $\mathfrak{a}$ is defined inductively in the usual way under the following semantics of temporal operators:

$n \models^{\mathfrak{a}} \bigcirc\varphi$	iff	$n+1 \models^{\mathfrak{a}} \varphi$;
$n \models^{\mathfrak{a}} \Diamond\varphi$	iff	there exists a $m \geq n$ such that $m \models^{\mathfrak{a}} \varphi$;
$n \models^{\mathfrak{a}} \Box\varphi$	iff	$m \models^{\mathfrak{a}} \varphi$ for all $m \geq n$;
$n \models^{\mathfrak{a}} \varphi\,\mathcal{U}\,\psi$	iff	there exists a $m \geq n$ such that $m \models^{\mathfrak{a}} \psi$ and for every $k \in \mathbb{N}$, if $n \leq k < m$ then $k \models^{\mathfrak{a}} \varphi$.

A formula φ is said to be *satisfiable* if there is a first-order structure $\mathfrak{M}$ and an assignment $\mathfrak{a}$ such that $\mathfrak{M}_0 \models^{\mathfrak{a}} \varphi$. If $\mathfrak{M}_0 \models^{\mathfrak{a}} \varphi$ for every structure $\mathfrak{M}$ and for all assignments $\mathfrak{a}$ then φ is said to be *valid*. Note that formulae here are interpreted in the initial state $\mathfrak{M}_0$; this is an alternative, but equivalent, definition to the one used in [HWZ00]. Let TL be the set of all valid $\mathcal{TL}$-formulae, and TL^- be the set of all valid $\mathcal{TL}$-formulae without equality.

Following [HWZ00] we consider the set of all $\mathcal{TL}$-formulae φ such that any subformula of φ of the form $\Diamond\psi$, $\Box\psi$, $\bigcirc\psi$, $\psi_1\,\mathcal{U}\,\psi_2$ has at most one free variable. Such formulae are called *monodic*, and the set of monodic $\mathcal{L}$-formulae is denoted by $\mathcal{TL}_1$. It was shown in [WZ01] that the set $TL \cap \mathcal{TL}_1$ is not recursively enumerable, while for the set $TL^- \cap \mathcal{TL}_1$ a finite Hilbert-style axiomatization has been provided.

It is reasonable to consider natural restrictions of $\mathcal{TL}_1$ corresponding to well-known fragments of the classical first-order language $\mathcal{L}$ such as monadic, two-variable and guarded fragments [HWZ00, WZ01]. Let $\mathcal{TL}_1^{mo}$, $\mathcal{TL}_1^2$ and $\mathcal{TL}_1^{\mathcal{G}}$ denote monadic, two-variable and guarded fragments of $\mathcal{TL}_1$, respectively. Let us recall that $\mathcal{TL}_1^{mo}$ consists of all $\mathcal{TL}_1$-formulae containing only unary predicates and propositional symbols, and $\mathcal{TL}_1^2$ contains all $\mathcal{TL}_1$-formulae with at most two variables. It was proved in [HWZ00] that each of the sets $TL^- \cap \mathcal{TL}_1^{mo}$, $TL^- \cap \mathcal{TL}_1^2$ and $TL^- \cap \mathcal{TL}_1^{\mathcal{G}}$ is decidable. Below we show that, after introducing equality, the sets $TL \cap \mathcal{TL}_1^{mo}$ and $TL \cap \mathcal{TL}_1^2$, together with their intersection $TL \cap \mathcal{TL}_1^{mo} \cap \mathcal{TL}_1^2$, become not even partially decidable, i.e. not recursively enumerable. Our proof is based on the argument that Minsky machines [Min67] can be simulated by formulae of $\mathcal{TL}_1^{mo} \cap \mathcal{TL}_1^2$. As to the guarded fragment $TL \cap \mathcal{TL}_1^{\mathcal{G}}$ the question of its decidability/enumerability remains open.

2 Minsky machines

The (two-counter) Minsky machine represents a universal model of computation [Min61, Min67]. Being of very simple structure, Minsky machines are very convenient for proving undecidability results (see for example [Hüt94, KR95, CZ97, GMV99]).

A Minsky machine $\mathcal{M}$ is a simple imperative program consisting of a sequence of instructions labelled by natural numbers from 1 to some L. It starts from an instruction labelled 1 and operates with two counters S_1 and S_2 each containing a nonnegative integer. Any instruction is one of the following forms:

l: ADD 1 to S_k; GOTO l';
l: IF $S_k \neq 0$ THEN SUBTRACT 1 FROM S_k; GOTO l' ELSE GOTO l'';
l: STOP.

where $k \in \{1, 2\}$ and $l, l', l'' \in \{1, \ldots, L\}$.

Without loss of generality one can suppose that every machine contains exactly one instruction of the form l: STOP which is the last one ($l = L$). It should be clear that the execution process (run) is deterministic and has no failure. Any such process is either finished by the execution of the STOP instruction or lasts forever.

As a consequence of the universality of such computational model the halting problem for Minsky machines is undecidable:

Theorem 1 ([Min67]). *It is undecidable whether a two-counter Minsky machine halts when both counters initially contain* 0*.*

We will use the following consequence of Theorem 1.

Corollary 1. *The set of all Minsky machines which begin with both counters containing* 0 *and do not halt is not recursively enumerable.*

Given any machine $\mathcal{M}$ (with initial values for the two counters) let us define its run $r^{\mathcal{M}}$ as a sequence of triples, or states of $r^{\mathcal{M}}$:

$$(l_1, p_1^0, p_2^0), (l_2, p_1^1, p_2^1), \ldots, (l_{j+1}, p_1^j, p_2^j), \ldots$$

where l_j is the label of the instruction to be executed at jth step of computation, p_1^j and p_2^j are the nonnegative integers within the first and the second counters, respectively, after completion of jth step of computation. Depending on whether $\mathcal{M}$ stops or not $r^{\mathcal{M}}$ can be finite or infinite.

Henceforth we will consider only the computations of the Minsky machines started with both counters containing 0. Thus we always put $p_1^0 = 0$, $p_2^0 = 0$ and $l_1 = 1$.

3 The reduction to the monodic monadic fragment

Given a Minsky machine $\mathcal{M}$ defined by the sequence of instructions $c_1, \ldots, c_L$ we define a first-order temporal formula $\chi^{\mathcal{M}}$ as follows.

Let P_1 and P_2 be unary predicate symbols. The intention is to model the contents of counters S_1 and S_2 during the computation by cardinalities of the extensions of P_1 and P_2, respectively, evolving in a temporal structure $\langle D, \mathcal{I} \rangle$. Here the extension (or the truth domain) of P_k at a moment n is $P_k^{\mathcal{I}(n)} = \{d \in D \mid P_k^{\mathcal{I}(n)}(d) = \textbf{true}\}$, $k = 1, 2$.

Let $Q_1, \ldots, Q_L$ be propositional symbols corresponding to instructions $c_1, \ldots, c_L$. Since we assume c_L is the STOP instruction we will denote Q_L alternatively as Q_{stop}.

Then, for every instruction c_l, except L: STOP, we define its translation $\chi(c_l)$ as follows:

A. An instruction of the form

l: ADD 1 to S_k; GOTO l'

is translated into the conjuction of the following formulae:

A1. $\Box(Q_l \to \forall x(P_k(x) \to \bigcirc P_k(x)))$
A2. $\Box(Q_l \to \forall x \forall y(\bigcirc P_k(x) \wedge \bigcirc P_k(y) \wedge \neg P_k(x) \wedge \neg P_k(y) \to x = y))$
A3. $\Box(Q_l \to \exists x(\neg P_k(x) \wedge \bigcirc P_k(x)))$
A4. $\Box(Q_l \to \forall x(P_{3-k}(x) \leftrightarrow \bigcirc P_{3-k}(x))$
A5. $\Box(Q_l \to \bigcirc Q_{l'})$

Formulae A1–A4 ensure that in every temporal model $\mathfrak{M}$ for them, once we have $Q_l^n =$ **true** at a moment n, at the next moment the extension of P_k should increase by one element, while the extension of P_{3-k} should be the same , i.e. $| P_k^{n+1} | = | P_k^n | +1, | P_{3-k}^{n+1} | = | P_{3-k}^n |$ $(k = 1, 2)$. The formula A5 describes switching truth values of propositions Q_i ($i \in \{1, \dots L\}$), and the aim here is to model the transition from the instruction which is executed to the next one.

B. An instruction of the form

l: IF $S_k \neq 0$ THEN SUBTRACT 1 FROM S_k GOTO l' ELSE GOTO l''

is translated into the conjunction of the following formulae:

B1. $\Box((Q_l \wedge \exists z P_k(z)) \to \forall x(\bigcirc P_k(x) \to P_k(x)))$
B2. $\Box((Q_l \wedge \exists z P_k(z)) \to \forall x \forall y(P_k(x) \wedge P_k(y) \wedge \bigcirc \neg P_k(x) \wedge \bigcirc \neg P_k(y) \to x = y))$
B3. $\Box((Q_l \wedge \exists z P_k(z)) \to \exists x(\bigcirc \neg P_k(x) \wedge P_k(x)))$
B4. $\Box((Q_l \wedge \exists z P_k(z)) \to \forall x(P_{3-k}(x) \leftrightarrow \bigcirc P_{3-k}(x)))$
B5. $\Box((Q_l \wedge \neg \exists z P_k(z)) \to \forall x(P_1(x) \leftrightarrow \bigcirc P_1(x) \wedge P_2(x) \leftrightarrow \bigcirc P_2(x)))$
B6. $\Box((Q_l \wedge \exists z P_k(z)) \to \bigcirc Q_{l'})$
B7. $\Box((Q_l \wedge \neg \exists z P_k(z)) \to \bigcirc Q_{l''})$

Formulae B1–B4 ensure that, in every temporal model for them, once we have Q_l and $\exists z P_k(z)$ true at some state, in the next state the extension of P_k should shrink by one element and the extension of P_{3-k} should remain the same. Formula B5 ensures that, when Q_l and $\neg \exists z P_k(z)$ holds, then, in the next state, the extensions of both P_1 and P_2 should remain the same. Formulae B6 and B7 regulate the switching of truth values of Q_i ($i \in \{1, \dots L\}$).

Denote by ρ the formula $\Box(\bigvee_{i=1}^{L} Q_i \wedge (\bigwedge_{i=1}^{L}(Q_i \to \bigwedge_{j \neq i} \neg Q_j)))$, stating that at every moment of time precisely one of Q_i is true.

Further, let the formula χ_0 be $Q_1 \wedge \rho \wedge \forall x(\neg P_1(x) \wedge \neg P_2(x))$ and let $\chi^{\mathcal{M}}$ be $\bigwedge_{i=1}^{L-1}(\chi(c_k))$ where $\mathcal{M}$ is a Minsky machine defined by the sequence of instructions $c_1, \dots, c_L$.

The formula $\chi_0 \wedge \chi^{\mathcal{M}}$ is intended to faithfully describe the computation of the machine $\mathcal{M}$ and the following lemma provides a formal justification for this.

Lemma 1. *A Minsky machine $\mathcal{M}$ produces an infinite run if, and only if, $\chi_0 \wedge \chi^{\mathcal{M}} \models \Box \neg Q_{stop}$.*

Proof

$\boxed{\Rightarrow}$ Let a machine $\mathcal{M}$ produce an infinite run

$$r^{\mathcal{M}} = (l_1, p_1^0, p_2^0), (l_2, p_1^1, p_2^1), \ldots (l_{j+1}, p_1^j, p_2^j), \ldots$$

and a temporal structure $\mathfrak{M} = \langle D, \mathcal{I} \rangle$ be a model of $\chi_0 \wedge \chi^{\mathcal{M}}$. We show by induction on steps in $r^{\mathcal{M}}$ that, for all $j \geq 1$, the following relation between states of $\mathcal{M}$ and $\mathfrak{M}$ holds:

$l_j = l$ whenever $j - 1 \models Q_l$;
$p_1^j = |P_1^j|$;
$p_2^j = |P_2^j|$.

For the base case, we have $0 \models Q_1$ and the label of the first instruction of $\mathcal{M}$ to be executed is $l_1 = 1$, extensions of both P_1 and P_2 are empty and values p_1^0 and p_2^0 of both counters at the beginning are 0. This establishes the base case.

For the step case assume for some j that $j - 1 \models Q_l$, $l_j = l$, $p_1^j = |P_1^j|$, $p_2^j = |P_2^j|$. Since the run $R^{\mathcal{M}}$ is infinite, the instruction c_l can be either of the first form, or of the second, but not the STOP instruction.

If an instruction is of the first form, i.e l: ADD 1 to S_k; GOTO l', then we have $l_{j+1} = l'$, $p_k^j = p_k^{j-1} + 1$ and $p_{3-k}^j = p_k^{j-1}$. Since the structure $\mathfrak{M}$ is a model of $\chi_0 \wedge \chi^{\mathcal{M}}$ and, in particular, a model of the translation of c_l, we have $P_k^j = P_k^{j-1} \cup \{d\}$ for some $d \notin P_k^{j-1}$, $P_{3-k}^j = P_{3-k}^{j-1}$ and $j \models Q_{l'}$. It follows that $l_{j+1} = l'$, $p_1^{j+1} = |P_1^{j+1}|$, $p_2^{j+1} = |P_2^{j+1}|$, as required.

The case of an instruction of the second form is considered in the same way, and translation of the instruction again ensures that the extensions of P_1, P_2 and the truth values of Q_i in the temporal structure $\mathfrak{M}$ model the values of counters and labels of current instruction, as above.
Thus, the step case is also established.

Since the run $r^{\mathcal{M}}$ is infinite we have $l_j \neq L$ for all $j \geq 1$, and therefore $j \models \neg Q_{stop}$ for all $j \geq 0$. Hence, $0 \models \Box \neg Q_{stop}$.

$\boxed{\Leftarrow}$ By contraposition it is sufficient to show that if a machine $\mathcal{M}$ produces a finite run (halts) then $\chi_0 \wedge \chi^{\mathcal{M}} \wedge \Diamond Q_{stop}$ is satisfiable.

Let a machine, $\mathcal{M}$, halt and produce a finite run $r^{\mathcal{M}} = (l_1, p_1^0, p_2^0), \ldots (l_{s+1}, p_1^s, p_2^s)$, $s \geq 0$. The final executed instruction is the STOP instruction, so we have $l_{s+1} = L$. Now, we construct a temporal structure $\mathfrak{M}_n = \langle D, \mathcal{I}(n) \rangle$ as follows. We let the domain D be a countable set. Then, for all $1 \leq j \leq s + 1$, we ensure $(j - 1) \models Q_l$ whenever $l_j = l$, and $j - 1 \models Q_{stop}$ for all $j > s + 1$. Further, we set $P_1^0 = P_2^0 = \emptyset$, and for all $1 \leq j \leq s$ define P_1^j and P_2^j inductively as follows:

- If the instruction with the label l_j is of the first form (ADD) then define $P_k^j = P_k^{j-1} \cup \{d\}$ where d is any element of D such that $d \notin P_k^{j-1}$, and $P_{3-k}^j = P_{3-k}^{j-1}$;
- If the instruction with label l_j is of the second form (SUBTRACT) and $(j - 1) \models \exists z P_k(z)$ then define $P_k^j = P_k^{j-1} \setminus \{d\}$, where d is any element of D such that $d \in P_k^{j-1}$, and $P_{3-k}^j = P_{3-k}^{j-1}$;

– If the instruction with the label l_j is of the second form (SUBTRACT) and $(j-1) \not\models \exists z P_k(z)$ then $P_1^j = P_1^{j-1}$ and $P_2^j = P_2^{j-1}$.

Finally, assume P_1^j and P_2^j to be arbitrary for all $j > s$.

It is easily seen that this overall construction provides a model for $\chi_0 \wedge \chi^{\mathcal{M}}$ and since $l_{s+1} = L$ one also has $s \models Q_{stop}$. Thus, $\chi_0 \wedge \chi^{\mathcal{M}} \wedge \Diamond Q_{stop}$ is satisfied in $\mathfrak{M}$.

Theorem 2. *The set $TL \cap \mathcal{TL}_1^{mo} \cap \mathcal{TL}_1^2$ consisting of all formulae of the monodic monadic two-variable fragment of first-order temporal logic with equality valid in temporal structures over the natural numbers is not recursively enumerable.*

Proof We note that the formulae of the form $\chi_0 \wedge \chi^{\mathcal{M}} \to \Box\neg Q_{stop}$ belong to $TL \cap \mathcal{TL}_1^{mo} \cap \mathcal{TL}_1^2$. Hence, the statement follows from Theorem 1 and Lemma 1.

4 Conclusion

In [WZ01] it is shown that the monodic fragment with equality is not recursively enumerable. At the same time the question of what occurs within the decidable monodic fragments of first-order temporal logic found in [HWZ00] after extending the language with equality is left open. In this paper we have shown that the monodic monadic two-variable fragment with equality is not recursively enumerable. Let us note that, in classical first-order logic, adding equality to monadic or two-variable fragments does *not* destroy their decidability [BGG97].

The proof of incompleteness of the monodic fragment with equality given in [WZ01] was based on the reduction to Craig's [Cra50] and Trakhtenbrot's [Tra50] result about incompleteness of the set of all first-order formulae valid in all finite interpretations. Roughly speaking, a formula $\chi \wedge \chi'$ of first-order temporal logic with equality was presented characterizing finite sets, and having this formula it was shown that the set of temporal formulae $(\chi \wedge \chi') \supset \varphi$, where φ is arbitrary classical first-order formula, is not recursively enumerable. The formula $\chi \wedge \chi'$, similar to our formula $\chi_0 \wedge \chi^{\mathcal{M}}$, is at once monodic, monadic and two-variable. However in [WZ01] there were no constraints related to φ. In our case a very simple temporal formula took the place of φ due to immediate simulation of Minsky machines. Taking into account further results on Trakhtenbrot's theorem (e.g. [Vua60]) it seems to be possible to restrict φ such that it would contain, besides monadic predicates, only one binary predicate. In such a way it would be possible to extend the proof of incompleteness of monodic logic with equality to the two-variable fragment, but not to the monadic case.

Let us note that there is a simpler temporal formula characterizing finite sets which is monodic, monadic and two-variable at the same time. We represent this by the conjunction of two formulae

1. $\Box\forall x\forall y((P(x) \wedge P(y)) \supset x = y)$,
2. $P(e) \wedge \bigcirc\forall y(\neg P(e)\,\mathcal{U}\,P(y))$

where e is a constant which can be replaced by the outermost existential quantifier. The first formula tells that, at any time point, there is at most one element of the domain for which the predicate P holds true. The second formula tells us that $P(e)$ is true at the initial moment, there is another time point where $P(e)$ has again to be true, and in the meantime, at successive time points, the predicate P has to be true for all other elements of the domain. As distances between time points are finite, the domain of any model must be finite as well. This formula is obtained from the formula without predicate symbols but with a flexible variable given in [Sza95] after replacing the flexible variable by a (flexible) monadic predicate symbol.[2]

Another interesting and important monodic fragment, for which decidability without equality was proved in [HWZ00], is the guarded fragment. Unfortunately neither $\chi \wedge \chi'$ nor $\chi_0 \wedge \chi^{\mathcal{M}}$ nor the new formula given above are guarded. So, the question about decidability/enumerability of the monodic and guarded fragment with equality is open as before.

Related papers dealing with undecidable guarded fragments of *non-temporal* first-order logic with added *transitive relations* are [GMV99] and [Grä99]. In [Grä99] it is shown that the three-variable guarded fragment equipped with two transitive binary relations is not recursively enumerable, while in [GMV99] the authors have shown that the two-variable guarded fragment without equality, but equipped with five transitive relations (or, with equality and four transitive relations), also becomes non recursively enumerable. (In the first article Trakhtenbrot's theorem is used, in the second article encoding Minsky machines has been applied.) On the other hand, the guarded non-temporal fragment with equality but without any additional relations is still decidable [AvBN96, AvBN98, Grä99, GdN99]. Thus, the case of the temporal monodic guarded fragment with equality can be seen as falling somewhere in between because after a standard translation into two-sorted first-order logic (see, e.g., [Aba89, HWZ00]) we obtain a syntactically restricted fragment with equality and with *one* linear order on natural numbers.

5 Acknowledgements

We would like to thank Michael Zakharyaschev for many helpful discussions. This work was supported by EPSRC under research grant GR/M46631.

References

[Aba89] M. Abadi. The power of temporal proofs. *Theoretical Computer Science*, 65(1):35–84, 1989.

[AvBN96] H. Andréka, J. van Benthem, and I. Németi. Modal languages and bounded fragments of predicate logic. Technical report, ILLC ML–96–03, 1996. 59 pages.

[AvBN98] H. Andréka, J. van Benthem, and I. Németi. Modal languages and bounded fragments of predicate logic. *Journal of Philosophical Logic*, 27:217–274, 1998.

[BGG97] E. Börger, E. Grädel, and Yu. Gurevich. *The Classical Decision Problem*. Springer, 1997.

[2] Reductions of the validity problems between logics with rigid predicates and flexible variables and logics with flexible predicates and without flexible variables were established in [Mer92].

[Cra50] W. Craig. Incompletability, with respect to validity in every finite nonempty domain, of first-order functional calculus. In *Proceedings of the International Congress of Mathematicians*, page 721, Cambridge, Mass., 1950.

[CZ97] A.V. Chagrov and M.V. Zakharyaschev. *Modal Logic*. Oxford Logic Guides 35. Clarendon Press, Oxford, 1997.

[Fis97] M. Fisher. A normal form for temporal logics and its applications in theorem-proving and execution. *Journal of Logic and Computation*, 7(4), 1997.

[GdN99] H. Ganzinger and H. de Nivelle. A superposition decision procedure for the guarded fragment with equality. In *Proceedings of 14th Annual IEEE Symposium on Logic in Computer Science(LICS'99)*, pages 295–303, 1999.

[GMV99] H. Ganzinger, C. Meyer, and M. Veanes. The two-variable guarded fragment with transitive realtions. In *Proceedings of 14th Annual IEEE Symposium on Logic in Computer Science(LICS'99)*, pages 24–34, 1999.

[Grä99] E. Grädel. The restraining power of guards. *Journal of Symbolic Logic*, 4:1719–1742, 1999.

[Hüt94] H. Hüttel. Undecidable equivalence for basic parallel processes. In *Proceedings of the International Conference on Theoretical Aspects of Computer Software TACS'94*, volume 789 of *Lecture Notes in Computer Science*, pages 454–464. Springer Verlag, 1994.

[HWZ00] I. Hodkinson, F. Wolter, and M. Zakharyaschev. Fragments of first-order temporal logics. *Annals of Pure and Applied logic*, 106:85–134, 2000.

[Kaw87] H. Kawai. Sequential calculus for a first order infinitary temporal logic. *Zeitschrift für Mathematische Logic and Grundlagen der Mathematik*, 33:423–432, 1987.

[KR95] G. Kucherov and M. Rusinowitch. Undecidability of ground reducibility for word rewriting systems with variables. *Information Processing Letters*, 53(4):209–215, 1995.

[Mer92] S. Merz. Decidability and incompleteness results for first-order temporal logic of linear time. *Journal of Applied Non-Classical Logics*, 2:139–156, 1992.

[Min61] M.L. Minsky. Recursive unsolvability of Post's problem of "tag" and other topics in theory of Turing machines. *Annals of Mathematics*, 74(3):437–455, 1961.

[Min67] M.L. Minsky. *Computation: Finite and Infinite Machines*. Prentice-Hall International, 1967.

[SH88] A. Szalas and L. Holenderski. Incompleteness of first-order logic with until. *Theoretical Computer Science*, 57:317–325, 1988.

[Sza86] A. Szalas. Concerning the semantic consequence relation in first-order temporal logic. *Theoretical Computer Science*, 47:329–334, 1986.

[Sza87] A. Szalas. A complete axiomatic characterization of first-order temporal logic of linear time. *Theoretical Computer Science*, 54:199–214, 1987.

[Sza95] A. Szałas. Temporal logic: A standart approach. In *Time and Logic. A Computational Approach*, chapter 1, pages 1–50. UCL Press Ltd., London, 1995.

[Tra50] B.A. Trakhtenbrot. The impossibility of an algorithm for the decision problem for finite models. *Dokl. Akad. Nauk SSSR*, 70:596–572, 1950. English translation in: *AMS Transl. Ser.* 2, vol.23 (1063), 1–6.

[Vua60] R. Vuaght. Sentences true in all constructive models. *Journal of Symbolic Logic*, 25:39–58, 1960.

[WZ01] F. Wolter and M. Zakharyaschev. Axiomatizing the monodic fragment of first-order temporal logic. To appear in *Annals of Pure and Applied logic.*, 2001.